Hindawi
Shock and Vibration
Volume 2023, Article ID 1830694, 14 pages
https://doi.org/10.1155/2023/1830694

Research Article

Remaining Useful Life Prediction of Milling Tool Based on Pyramid CNN

Ning Hu,[1,2] **Zhenguo Liu**,[1,2] **Shixin Jiang**,[1,2] **Quanzhou Li**,[1,2] **Shuqi Zhong**,[1,2] **and Bingquan Chen**[1,2]

[1] *China Electronic Product Reliability and Environmental Test Institute, Guangzhou 510610, China*
[2] *Key Laboratory of Industrial Equipment Quality Big Data, MIIT, Guangzhou 510610, China*

Correspondence should be addressed to Shixin Jiang; jiangshixinwh@163.com

Received 16 June 2022; Revised 18 October 2022; Accepted 24 November 2022; Published 11 February 2023

Academic Editor: Liu Jing

Remaining useful life prediction of a milling tool is one of the determinants in making scientific maintenance decision for the CNC machine tool. Predicting the RUL accurately can improve machining efficiency and the quality of product. Deep learning methods have strong learning capability in RUL prediction and are extensively used. Multiscale CNN, a typical deep learning model in RUL prediction, has a large number of parameters because of its parallel convolutional pathways, resulting in high computing cost. Besides, the MSCNN ignores various influences of different scales of degradation features on RUL prediction accuracy. To address the issue, a pyramid CNN (PCNN) is proposed for RUL prediction of the milling tool in this paper. Group convolution is used to replace parallel convolutional pathways to extract multiscale features without additional large number of parameters. And the channel attention with soft assignment is used to select the key degradation features, considering different sensors and scales. The milling tool wear experiments show that the score value of the proposed method achieved 51.248 ± 1.712 and the RMSE achieved 19.051 ± 0.804, confirming better performance of the proposed method compared with the traditional MSCNN and other deep learning methods. Besides, the number of parameters of the proposed method is reduced by 62.6% and 54.8% compared with the MSCNN with self-attention and the MSCNN methods, confirming its lower computing cost.

1. Introduction

As a basic tool of industry, computer numerical control (CNC) machine tool plays an important role in industrial manufacture. With the increasing demand for product quality, stability of machining process becomes more and more important. Tool wear is a common negative effect on machining quality during the high-speed machining process [1]. And it not only affects the quality of machined surface and the machining precision but also results in increasing machining cost. Moreover, unnecessary tool replacement that aims at preventing the decrease in surface quality will increase the downtime and machining cost in high-speed milling [2]. The effects for tool degradation mainly include cutting parameters, work material, and cutting tool. However, the internal law of these effects on tool degradation is hard to determine for their various combinations. Since it could not be directly detected during the process, it is hard to make scientific maintenance decisions without interrupting the machining process. Therefore, a significative work is to accurately predict the remaining useful life (RUL) of the milling tool.

With the widely usage of industrial internet of thing in condition monitoring of machinery, a mass of monitoring data of the CNC machine tool are acquired by various sensors. The explosive growth of monitoring data brings new opportunities to RUL prediction of the milling tool. Compared with model-driven RUL prediction methods, data-driven RUL prediction methods are able to learn degradation characteristics of a tool from massive monitoring data. And it could also build the corresponding RUL prediction models automatically, which means neither deep understanding of system-failure physics nor complete knowledge of the dynamics is required. Therefore, data-driven RUL prediction methods are gaining more and more attention in the field of RUL prediction recently [3].

Traditional data-driven prognostic approaches usually contain three steps: hand-crafted feature extraction, degradation behavior learning, and RUL prediction [4, 5]. Hand-crafted feature extraction is to use signal process methods to extract sensitive degradation features from the monitoring data. Then, these features are fed into machine learning models, such as ridge regression, support vector machine (SVM), and so on, to learn the degradation features and predict the RUL. For example, Park et al. [6] extract time, frequency, and time-frequency domain features, and these features are input into the ridge regression model after dimension reduction using PCA. Zhao et al. [7] extract high-dimensional feature using time-frequency representation (TFR), which are fed into the simple multiple linear regression model to predict the RUL after supervised dimensionality reduction using PCA and LDA. Liu et al. [8] used the integration of empirical mode decomposition (EMD) and Wigner–Ville distribution (WVD) to extract degradation feature from gearbox vibration signal, and then particle filter (PF) with the state space model based on the Wiener process is used to predict the RUL of gearbox considering degradation feature. Even though these methods have a good performance on the RUL prediction, they still need to take much effort on hand-crafted feature design [9, 10]. To avoid this situation, it is desirable to find a new method to automatically extract degradation feature from monitoring data. Therefore, deep learning-based RUL prediction methods have gained more and more attention in the field of data-driven RUL prediction [11–20].

Deep learning, structured by a stack of multiple layers of nonlinear processing units [21], can extract high-level feature without human intervention. Thus, deep learning shows a more powerful feature extraction ability, and achieves state-of-the-art accuracy in many tasks, such as image classification, natural language processing (NLP), target detection, and so on. Deep belief network (DBN), auto-encoder network (AEN), recurrent neural network (RNN), and convolution neural network (CNN) are mainstream architectures in deep learning [22]. Wang et al. [23] proposed a deep separable convolution network (DSCN) for RUL prediction of bearing, which extracted the degradation feature from monitoring data using deep separable convolution and predicted the RUL using fully connected layers. Hinchi and Tkiouat [24] used CNNS to extract features from vibration signal, and then employed LSTM to predict the RUL of rolling element bearings. Zhang et al. [25] proposed a multiobjective DBN ensemble method for RUL prediction of turbofan engines. Wang et al. [26] use DCAE and SOM to gain the health index of rolling bear, and then use this health index as a label to train a CNN-based RUL prediction model to predict the RUL. Ding et al. [27–29] proposed three meta deep learning methods to predict the RUL of the machine under different conditions and limited and variable-length data. Zhang et al. [30] proposed a deep representation regularization-based transfer learning method for remaining useful life predictions under different machinery operating conditions and no target-domain run-to-failure training data.

Because of the remarkable ability of extracting degradation features from monitoring data, CNN-based RUL prediction methods become a research hotspot, especially the multiscale CNN (MSCNN) [31–39]. The architecture of traditional MSCNN with self-attention is shown in Figure 1. Parallel convolutional pathways are used to extract different scales of degradation features, which is developed by different size of convolution kernel for different convolutional pathways. And the self-attention is embedded to avoid the interference caused by the redundant and uncorrelated information of partial sensors, improving the performance of the networks. The usage of parallel learning strategy, however, greatly increases the parameters of the model, leading to higher cost of computing during model training. The self-attention, in addition, can only consider the contribution of different sensors to RUL prediction. In other words, the contribution of different scale of degradation features is not taken into account.

To deal with the mentioned problems, a pyramid CNN (PCNN) is proposed in this paper. The architecture of the proposed PCNN is shown in Figure 2. The monitoring data acquired from different sensors can be directly fed into the proposed network without any preprocessing, which means complex signal processing techniques do not require. This network contains two parts, multiscale feature learning subnetwork and RUL predicting subnetwork. The multiscale feature learning subnetwork is built by stacking one-dimensional (1D) convolution layers and pyramid convolution layers. Low-level features are extracted by the one-dimensional (1D) convolution layers and fed into the pyramid convolution layers. In the pyramid convolution layers, group convolution is used to extract multiscale high-level degradation features. Then, the channel attention model is used to generate attention weight for each channel. A soft assignment is used to recalibrate the attention weight of different scales so that the key degradation features can be selected from not only different sensors but also from different scales. The RUL predicting subnetwork contains global pooling and fully connected layers (FCLs). The mapping relationship between degradation features and the RUL is established in these parts. The tool wear experiment is used to verify the proposed method. Compared with the traditional MSCNN, the proposed method has higher accuracy of RUL prediction and smaller number of parameters.

The rest of this article is structured as follows. The basic theory of the proposed method is expounded in detail in Section 2. Experiment and comparison analyses are illustrated in Section 3. Conclusions are composed in Section 4.

2. Proposed PCNN for RUL Prediction of Milling Tool

2.1. One-Dimensional (1D) Convolution Layer and Shortcut Connection. On-dimensional convolution is used to extract degradation feature from raw data in this paper. The 1-*D* convolutional operation can be described as

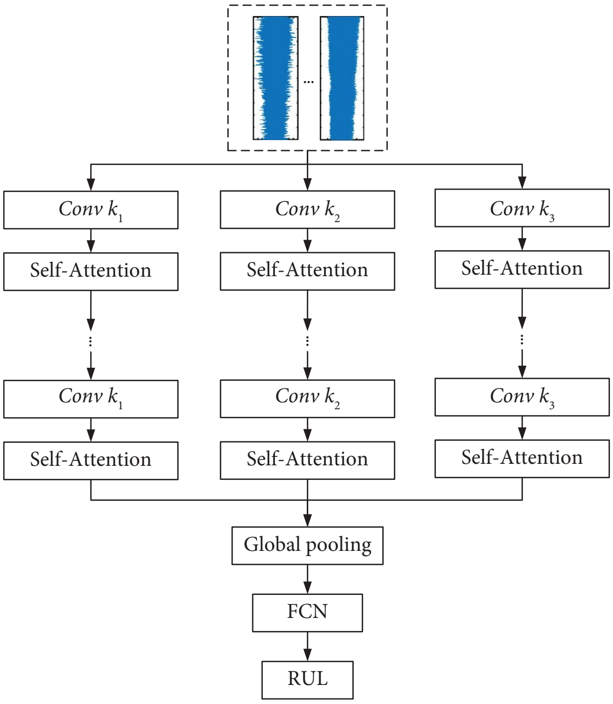


FIGURE 1: The architecture of traditional MSCNN with attention mechanism.

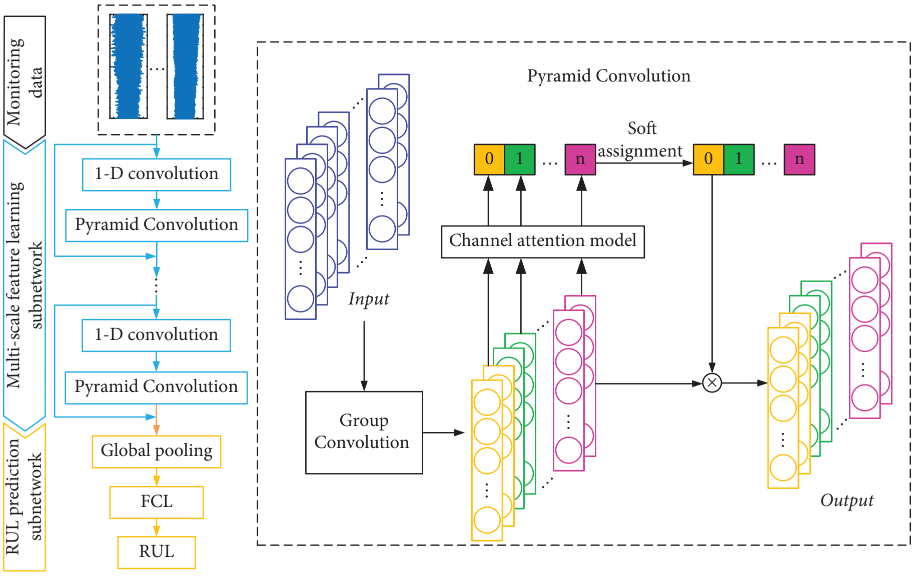


FIGURE 2: The architecture of the proposed PCNN.

$$y_0 = f\left(\sum_{i=1}^{n} k_{0,i} * X_0 + b_{0,i}\right), \qquad (1)$$

where X_0 is the raw data, y_0 is the output of the process, $k_{0,i}$ is the learnable convolutional kernel, $b_{0,i}$ is the bias tern, $*$ represent the convolutional operation, and $f(\cdot)$ is the nonlinear activation function. In this paper, the rectified linear unit (ReLU) is used as the nonlinear activation function of the 1-D convolution operation. By repeating this process twice, low-level degradation features, denoted as F_0, can be obtained.

Gradient vanishing/exploding and weight matrix degradation is a considerable problem of deep learning. To address this issue, shortcut connection is introduced in this network.

The raw data acquired from the sensor is fed into the shortcut connection pathway, which contains a convolution layer and a max pooling layer. The size of the convolutional kernel in the shortcut connection is 1 × 1, which aims to increase the dimension of X_0. The max pooling layer is used to downsample the output of the convolution layer. The output of the shortcut connection model, denoted as S_{out}, is given by

$$S_{\text{out}} = Y + \text{pool}\left(k_c * X_0\right), \tag{2}$$

where Y is the output of the pyramid convolution layer, $\text{pool}(\cdot)$ is the pooling function, k_c is the convolution kernel with the size of 1×1, and $*$ is the convolution operation.

2.2. Pyramid Convolution Layer. In this layer, multiscale high-level degradation information from different sensors is extracted and fused. First, a group convolution operation is used to extract different scale of high-level degradation features. After doing this, the channel attention model is used to generate the attention weights of the multiscale features. Finally, the soft assignment is used to recalibrate the attention weight of the corresponding scale.

2.2.1. Group Convolution. The monitoring data acquired from the sensors are nonlinear signals containing a lot of noise. While the degradation features can be extracted by convolution operation, the receptive field range of the convolution kernels have great influence on the degradation features. Large-scale degradation features can be extracted by a larger receptive field, while detailed degradation features can be extracted by a smaller receptive field. Therefore, it is necessary to use different size of convolution kernels to extract multiscale degradation features. The traditional multiscale convolution uses parallel pathways to extract multiscale features. The size of convolution kernel in various convolution pathways is different. Although the performance of the network is proved, a large number of parameters increases the computing cost. Therefore, it is desirable to find an efficient multiscale feature extraction method.

In this paper, group convolution is used to replace parallel convolutional pathways so that multiscale features can be extracted without additional large number of parameters. The architecture of this model is shown in Figure 3.

The input low-level feature $F_0 \in \mathbb{R}^{L\times C}$ is splitted into s groups along with the channel direction, denoted as $X_i \in \mathbb{R}^{L\times c/s}$, with $i = 1, 2, \ldots, s$, where c is the number of channel and L is the length of F_0. A set of learnable kernels is used to convolve X_i. The output of the convolution, denoted as F_i, can be obtained by

$$F_i = \sum_{c=1}^{\frac{C}{s}} k_{i,c} * X_i + b_{i,c}, \tag{3}$$

where C/s is the number of learnable kernels and the number of input channels, $*$ denotes the convolution operator, $k_{i,c} \in \mathbb{R}^{F\times 1\times (C/s)\times (C/s)}$ is the $c-th$ convolution kernel of the $i-th$ group, and $b_{i,c}$ is the bias term. Different convolution kernels $k_{i,c}$ have different sizes, which can extract different scales of degradation features. Finally, the whole multiscale feature can be obtained by the concatenation of all the F_i.

2.2.2. Channel Attention Model and Soft Assignment. The data from different sensors contain different degrees of degradation information. In other words, some important degradation information only exists in partial sensors. Furthermore, different scales of features also contain different degrees of degradation information. Therefore, it is important to select key degradation information from the multiscale feature F. In this paper, a channel attention model is used to obtain the attention weight from the input feature F. Then, the soft assignment is used to recalibrate the attention weight of the corresponding scale. The structure of this model is shown in Figure 4.

Attention weights of the features of different scales can be obtained by using parallel processing pathways. Each processing pathway includes global information encoding and channel-wise relationship information recalibrating. The global information encoding is done by global average pooling and global max pooling, and the channel-wise relationship information recalibrating is done by fully connected networks with one hidden layer.

The global average pooling (GAP) and the global max pooling (GMP) can aggregate the global information of each channel, generating two vectors: V_a and V_m. Both V_a and V_m contain $J = C/s$ channel-wise statistics. The channel-wise statistics of the j-th channel $V_{a,j}$ and $V_{m,j}$ is obtained by

$$V_{a,j} = \frac{1}{P}\sum_{p}^{P} V_{a,j,p} V_{m,j} = \max\left(V_{m,j,p}\right). \tag{4}$$

Then, V_a and V_m are fed into the fully connected network (FCN) with one hidden layer. The neuron number of the hidden layer in the FCN is J/r, where r is the ratio of dimensionality reduction. After that, the attention weight of F_i, denoted as Z_i, can be calculated by

$$Z_i = \sigma\left(W_{a2}\left(W_{a1}V_a\right) \oplus W_{v2}\left(W_{v1}V_m\right)\right), \tag{5}$$

where $W_{a1} \in \mathbb{R}^{J/R\times J}$, $W_{a2} \in \mathbb{R}^{J\times J/R}$, $W_{\text{v}1} \in \mathbb{R}^{J/R\times J}$, and $W_{\text{v}2} \in \mathbb{R}^{J\times J/R}$ are the weight matrices in the FCNs, $\oplus$ denotes the element-wise summation, and $\sigma(\cdot)$ is the sigmoid activation function.

By doing this, the network can fuse degradation information from different sensors and produce a better attention for high-level degradation feature. Furthermore, in order to enhance the key degradation features of some scales and suppress the irrelevant ones without destroying the original channel attention vector, a soft assignment is used to adaptively recalibrate the attention weight of the corresponding scale. After doing this, the key degradation features are selected not only from different sensors but also from different scales. The soft assignment is given by

$$\text{att}_i = \frac{\exp\left(Z_i\right)}{\sum_{i=o}^{s-1}\exp\left(Z_i\right)}. \tag{6}$$

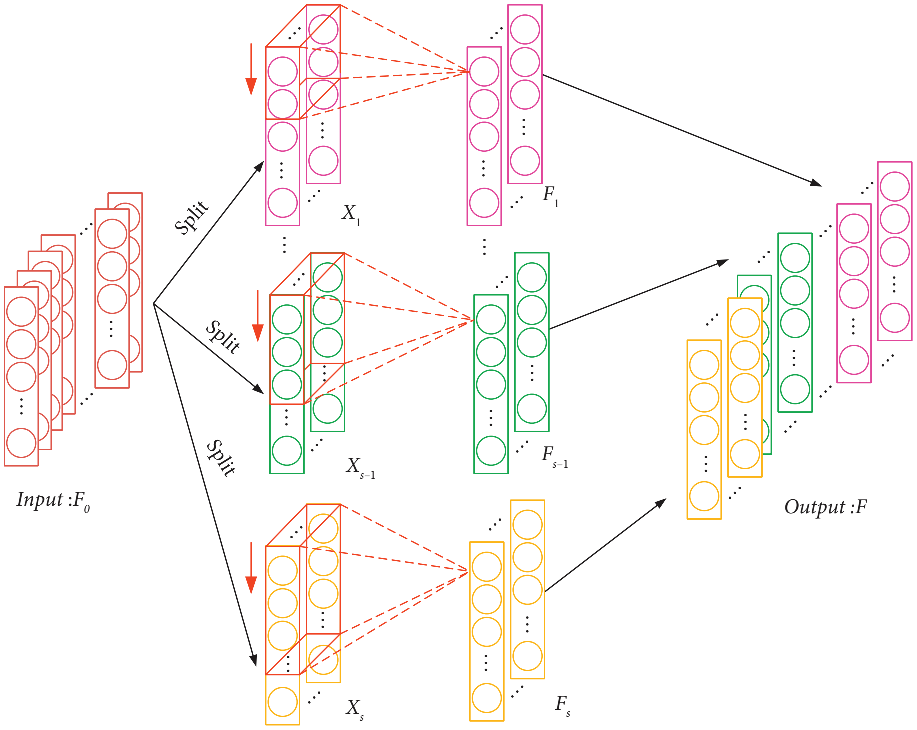


FIGURE 3: Architecture of the group convolution.

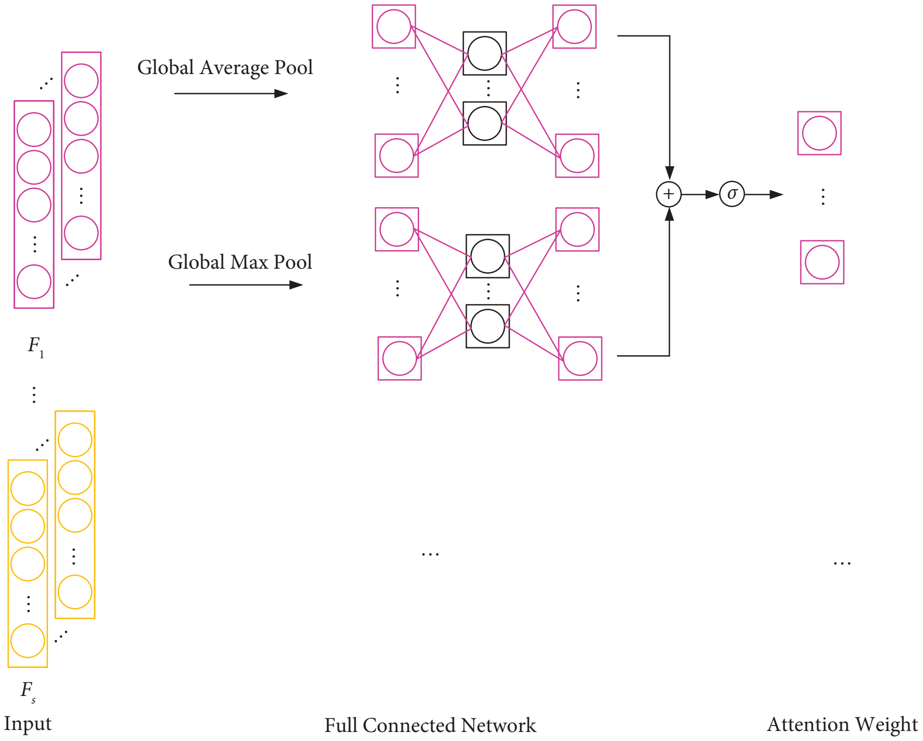


FIGURE 4: Architecture of the channel attention model.

Then, the multiscale high-level degradation feature with multiscale channel-wise attention weight, denoted as Y_i, can be obtained by

$$Y_i = F_i \odot \text{att}_i, \tag{7}$$

where $\odot$ is the channel-wise multiplication.

Finally, the output of the pyramid convolution layer, denoted as Y, can be obtained by the concatenation of all the Y_i.

3. Experimental Verification

3.1. Data Description. As shown in Figure 5, the life testing of the milling tool is conducted in a computer numerical control (CNC) milling machine.

The material of the workpiece is 316L stainless steel, and the milling tool is cemented carbide insert deposited by TiAlN coating. During the milling process, the table feeds the workpiece from front to back along the Y-axis. As tabulated in Table 1, a total of 4 milling tool are tested and all tests are carried out without the application of a cutting fluid. As shown in Figure 6, two types of sensors are installed in the milling machine, including accelerometer (Kistler Z292A600) and rotary dynamometer (Pro-Micro). For the accelerometer, the sampling frequency is set as 10 kHz. For the rotary dynamometer, the sampling frequency is set as 2.5 kHz.

As shown in Figure 6, a metallographic microscope is used to measure the width of the flank wear. When the width of the flank wear is greater than 0.2 mm, the tested tool wear achieves the limit [1]. The acquired monitoring data of the C1 during the whole operating life is shown in Figure 7.

As shown in Figure 7, some of these monitoring data have obvious degradation trends with the increasing of cutting time, while others do not have these trends.

3.2. Experimental Study. In this case, all of the monitoring data are used as the input of the network to verify the effectiveness of the proposed method. The size of an input sample is $10000 \times 1 \times 5$.

One of the main hyperparameters that may affect the prediction performance of the proposed model is the number of groups, which directly affects the dimension of feature extract in the pyramid convolution layer. For investigating this influence, different number of groups in the proposed PCNN are applied to estimate the RUL prediction. The number of groups is set to be 2, 4, and 8. Figure 8 shows the score values and RMSE of *C*4, and the corresponding training time and model parameters are given in Table 2.

It can be observed that the score value is the lowest and the RMSE is the highest when the number of group is set to be 2, which indicates that the prediction performance is relatively poor. The accuracy of the RUL prediction results is closer for others. As the number of groups increased, the model becomes more computationally intensive. Therefore, it can be observed in Table 2 that the model training time and the number of parameters increased with the increase in the number of groups. Though a bigger number of groups can extract more features of different scales, resulting in better prediction performance, the calculation burden is aggravated and the performance improvement is limited when the number of group increases to a certain extent. By the tradeoff between accuracy and efficiency, the number of groups is finally selected as 4.

The final architecture of the network is shown in Figure 9. And the hyperparameters of the pyramid convolution layer of the PCNN are listed in Table 3.

Mean square error is used as the loss function of the network and Adam optimizer with a mini-batch size of 128 is used to update its weights and biases. The trained network is used to predict the RUL values of the testing dataset after training 150 epochs. If the prediction value was bigger than the actual value, it may cause low process quality or even a scrapped products due to a overwear in the tool. Taking this situation into account, except for root mean square error (RMSE), a score function is used to evaluate the performance of the network. The score value is given by

$$\text{Score} = \frac{1}{S}\sum_{i=1}^{S} S_i,$$
$$S_i = \begin{cases} 100 * \exp^{-\ln(0.5)\cdot(y-\hat{y}/5),} & y \leq \hat{y}, \\ 100 * \exp^{\ln(0.5)\cdot(y-\hat{y}/20),} & y > \hat{y}, \end{cases} \tag{8}$$

where S is the number of samples in the testing dataset, y is the actual value, and $\hat{y}$ is the predicted value. The higher the score values, the more accurate the performance of the RUL prediction is.

Figure 10 shows the RUL prediction result of *C*4 using the proposed method. As shown in Figure 10, the predicted RUL value fluctuates slightly with the actual RUL, and the fluctuation becomes smaller and smaller with the increase of the cutting time. Furthermore, cross validation is used to prove the stability of the proposed method. Each test is repeated ten times, and the mean and standard deviation of these four testing dataset are listed in Table 4.

As shown in Table 4, on the one hand, both score and RMSE of each testing dataset has small standard deviation, which proves that the proposed model has good stability for the same task. On the other hand, the mean value of both score and RMSE of these four testing dataset has small fluctuation, which proves that the proposed network has good stability for different tasks. In conclusion, the proposed network has a good prediction result and good stability in both the same task and the different task, which means the predicted result of the proposed method is credible.

3.3. Comparison Analysis

3.3.1. Ablation Experiments. In order to illustrate the advantage of the proposed PCNN, ablation experiments are done in this part. The other three prognostic networks are employed to predict the RUL and they are denoted as Network-1, Network-2, and Network-3. The architectures of these three networks are similar to that of the PCNN, and the differences are that (1) Network-1 does not use group convolution and channel attention with soft assignment, (2) Network-2 only use group convolution, and (3) Network-3 only use channel attention with soft assignment. In addition, the hyperparameters settings of these three networks are the same as those of the PCNN, and the cross validation used in

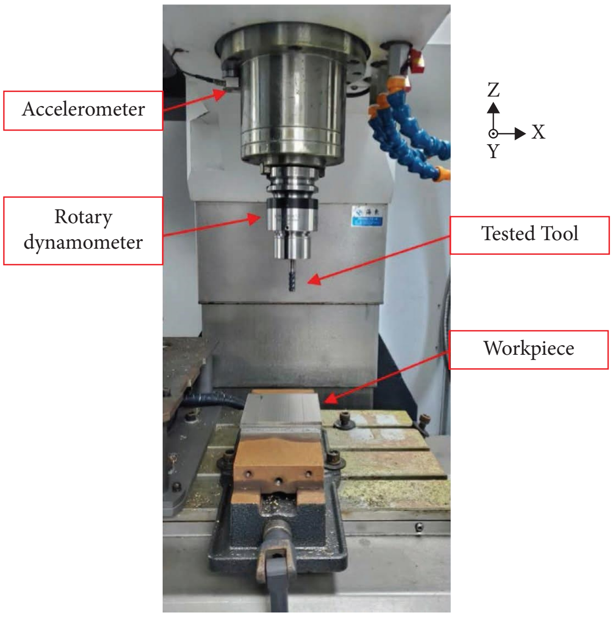


Figure 5: CNC machine and sensor placement.

Table 1: Cutting condition of milling tool.

Spindle speed	Feed rate	Depth of cut	Width of cut	Dataset
3500 rpm	300 mm/min	2 mm	2 mm	$C1$, $C2$, $C3$, and $C4$

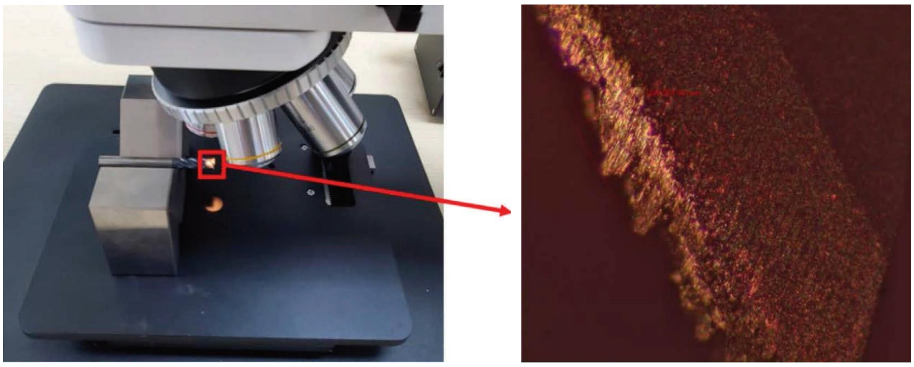


Figure 6: Milling tool deterioration photograph.

Section 3.2 is used in this part too. The performance estimation results of these four different networks are listed in Table 5 and drawn in Figure 11.

It can be observed that compared with the classic multiscale convolutional network without attention mechanisms (i.e., Network-1 [37]), the use of group convolution or channel attention with soft assignment effectively improves the prediction performance and stability of the network, resulting in higher score value and lower RMSE. For Network-2, the performance improvement is attributed to the use of group convolution, which reduces the risk of overfitting by reducing the number of learning parameters. For Network-3, the employment of channel attention with soft assignment make the network enhance key degradation features of some sensors and scales. Besides, it is to be noted that through systematically integrating group convolution and soft attention with soft assignment, the proposed PCNN obtains the highest score value and the lowest RMSE value for each testing dataset among four different prognostic networks, which verifies again the performance of the proposed method.

3.3.2. Comparison with the State-of-the-Art Models. In this part, eight state-of-the-art models, including two machine learning models, random forests (RF), and support vector

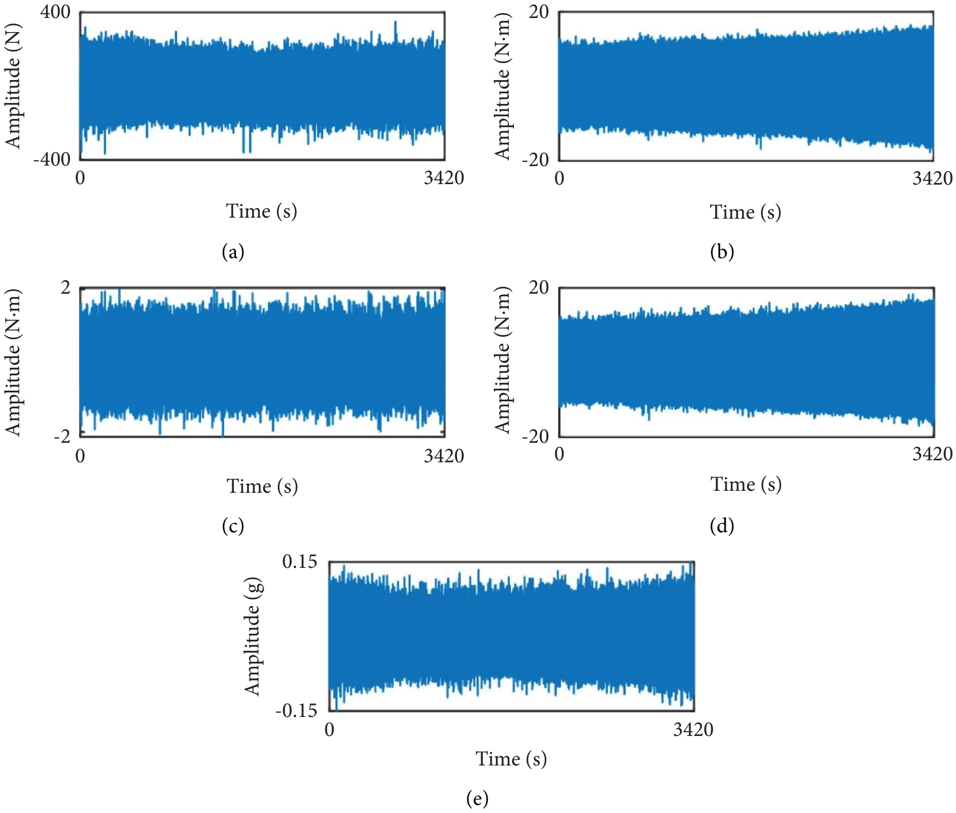


Figure 7: Monitoring data of the *C*1 during the whole operating life. (a) Force data in the *Z*-axis. (b) Bending moment data in the *X*-axis. (c) Torque data. (d) Bending moment data in the *Y*-axis. (e) Vibration data in the *Y*-axis.

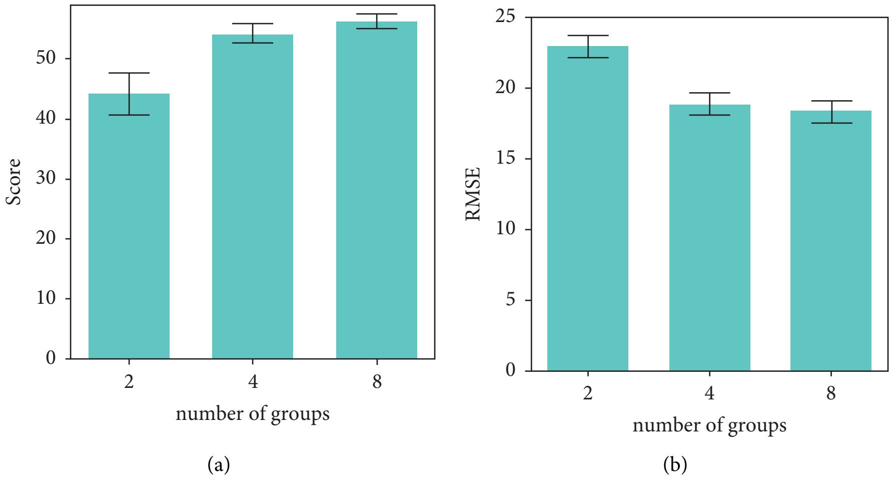


Figure 8: RUL prediction result of *C*4 based on different number of groups. (a) Score values. (b) RMSE.

regression (SVR) [34] and six deep learning model, deep convolution neural network (DCNN) [35], residual dense network (RDN) [36], multiscale convolutional neural network (MSCNN) [37], convolutional long-short-term memory network (CLSTM) [24], deep belief networks (DBN) [38], and multiscale convolutional attention network (MSCAN) [39] are utilized to estimate the RUL for the comparison analysis. For the RF and SVR, features listed in [34] are extracted from all the monitoring data. Then, these features are fed into the corresponding model to predict the RUL. The score value and RMSE of these methods are listed in Table 6. Both score value and RMSE are calculated form the half of the life too.

From Table 6, it can be found that the proposed method has the highest score value and the lowest RMSE, which confirms the proposed method can predict the RUL

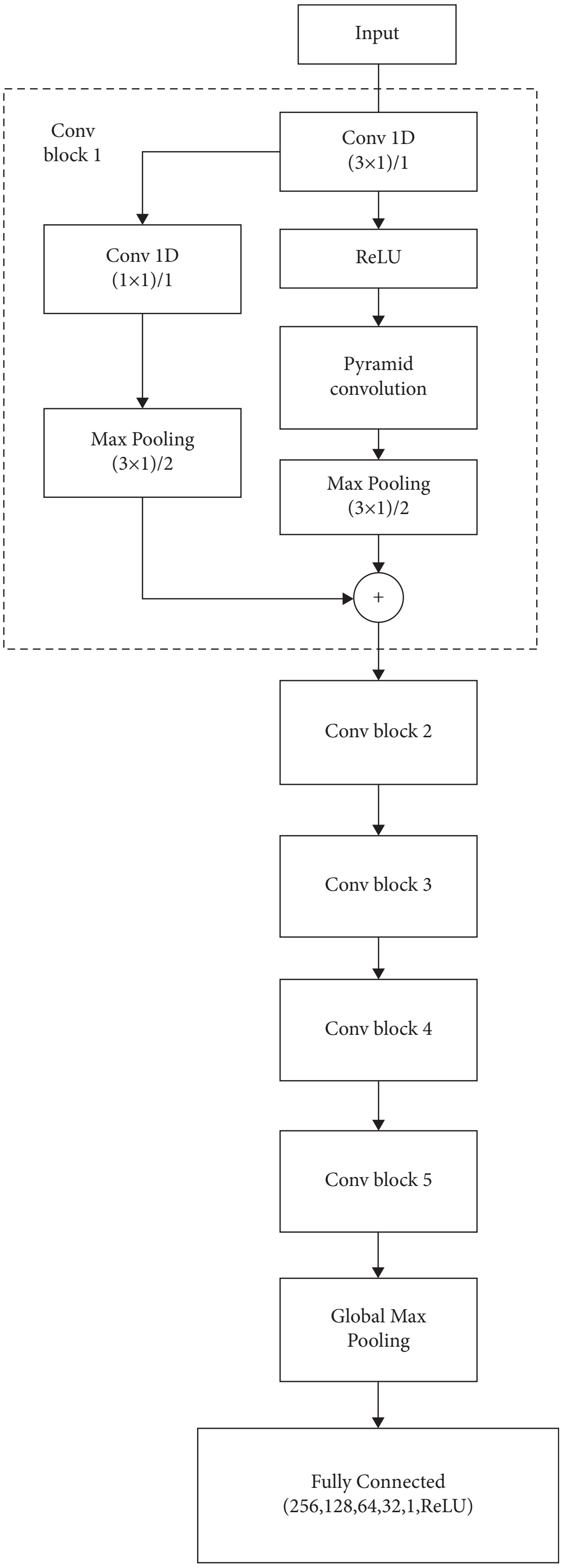


Figure 9: Architecture of the proposed PCNN.

Table 2: Comparison of model parameters and training time with different numbers of groups.

Number of groups	2	4	8
Training time (s)	1100	1230	1353
Total model parameters	613,225	765,993	1,115,425

Table 3: Hyperparameters of the pyramid convolution layer.

Hyperparameters	Values
The ratio of dimensionality reduction r	4
Size of convolution kernel k_i with $i = 1, 2, \ldots, s$	$3 \times 1,\ 5 \times 1,\ 7 \times 1,\ 9 \times 1$
Number of group s	4

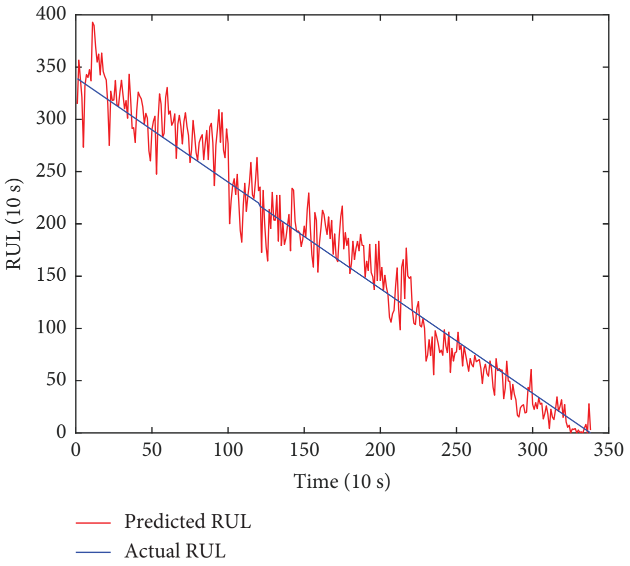


Figure 10: RUL prediction result of $C4$ using the proposed method.

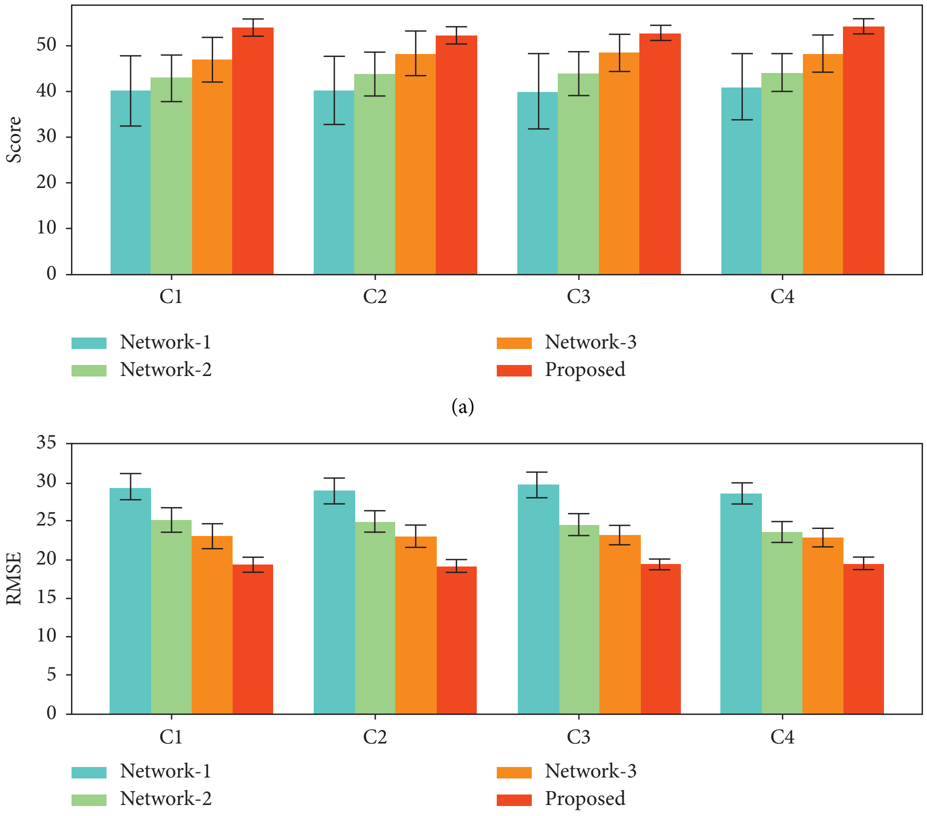
Table 4: Performance estimation result of four testing dataset.

Testing datasets	Scores	RMSE
C1	50.962 ± 1.813	19.374 ± 0.923
C2	51.217 ± 1.872	19.139 ± 0.853
C3	50.771 ± 1.617	19.424 ± 0.721
C4	51.248 ± 1.712	19.051 ± 0.804

Table 5: Performance estimation result of four different networks.

Testing datasets		Network-1	Network-2	Network-3	PCNN
C1	Score	40.152 ± 7.486	42.934 ± 5.063	47.013 ± 4.828	**53.962 ± 1.813**
	RMSE	29.407 ± 1.712	25.122 ± 1.592	23.114 ± 1.581	**19.374 ± 0.923**
C2	Score	40.274 ± 7.397	43.771 ± 4.811	48.167 ± 4.765	**52.217 ± 1.872**
	RMSE	28.903 ± 1.664	24.913 ± 1.428	23.022 ± 1.412	**19.139 ± 0.853**
C3	Score	39.914 ± 8.152	43.912 ± 4.702	48.369 ± 4.105	**52.771 ± 1.617**
	RMSE	29.729 ± 1.677	24.502 ± 1.437	23.216 ± 1.241	**19.424 ± 0.721**
C4	Score	40.889 ± 7.109	44.068 ± 4.155	48.154 ± 3.907	**54.248 ± 1.712**
	RMSE	28.597 ± 1.402	23.662 ± 1.339	22.901 ± 1.209	**19.051 ± 0.804**

The bold values express that the PCNN has the best performance.



(a)

(b)

Figure 11: Performance estimation result of four different networks. (a) Score values. (b) RMSE values.

accurately. This performance enhancement demonstrates again the advantage of the PCNN.

Besides, in order to illustrate the efficiency of the PCNN, the number of parameters and the training and testing time of three multiscale learning models are listed in Table 7. All experiments in this paper are performed on a server configured with two Intel (R) Xeon (R) Gold 6242R CPU@ 3.10 GHz processors, eight NVIDIA GeForce RTX 3090 graphics cards, and a total of 512 GB memory (RAM).

As shown in Table 7, the total model parameters of the proposed method are respectively reduced by 62.6% and 54.8% compared to the MSCNN with self-attention and the MSCNN methods. Both training time and testing time of the proposed method are greatly reduced, which means

Table 6: Performance estimation result of the testing dataset for eight state-of-the-art models.

Methods	Scores				RMSE			
	*C*1	*C*2	*C*3	*C*4	*C*1	*C*2	*C*3	*C*4
DCNN [35]	5.572 ± 10.297	4.835 ± 10.112	5.214 ± 10.822	5.713 ± 10.640	44.694 ± 2.558	45.206 ± 2.353	44.967 ± 2.301	44.515 ± 2.140
SVR [34]	16.997 ± 9.125	16.935 ± 9.002	17.279 ± 8.910	17.529 ± 8.812	39.012 ± 1.213	39.245 ± 1.221	38.914 ± 1.208	38.839 ± 1.194
DBN [38]	38.114 ± 5.198	38.202 ± 5.109	38.883 ± 4.950	39.163 ± 4.792	30.891 ± 1.315	31.048 ± 1.382	31.008 ± 1.213	30.427 ± 1.290
RDN [36]	31.341 ± 9.914	31.840 ± 9.757	32.019 ± 9.857	32.498 ± 9.309	29.890 ± 1.811	29.784 ± 1.887	29.560 ± 1.891	29.216 ± 1.921
MSCNN [37]	40.152 ± 7.486	40.274 ± 7.397	39.914 ± 8.152	40.889 ± 7.109	29.407 ± 1.712	28.903 ± 1.664	29.729 ± 1.677	28.597 ± 1.402
CLSTM [24]	45.331 ± 4.980	45.519 ± 4.832	45.712 ± 4.962	45.870 ± 4.954	28.245 ± 1.290	28.552 ± 1.285	28.771 ± 1.297	28.245 ± 1.314
MSCAN [39]	32.497 ± 2.244	32.575 ± 2.225	32.573 ± 2.294	32.814 ± 2.106	28.612 ± 1.104	28.504 ± 1.123	28.482 ± 1.109	28.374 ± 1.011
RF [34]	40.126 ± 2.268	40.125 ± 2.740	40.219 ± 2.442	40.370 ± 2.589	21.441 ± 1.071	21.372 ± 1.096	21.457 ± 1.101	21.371 ± 1.002
Proposed	**53.962 ± 1.813**	**52.217 ± 1.872**	**52.771 ± 1.617**	**54.248 ± 1.712**	**19.374 ± 0.923**	**19.139 ± 0.853**	**19.424 ± 0.721**	**19.051 ± 0.804**

The bold values express that the proposed method has the best performance.

Table 7: The number of parameters of different models.

Methods	Number of parameters	Training time/s	Testing time/s
MSCAN	2,047,409	1980	7
MSCNN	1,694,337	1838	6
Proposed	**765,993**	**1230**	**3**

the computing cost is reduced and the efficiency is improved.

4. Conclusion

Because of the strong learning capability, the CNN is widely used in degradation feature extraction, especially the multiscale CNN which has a stronger representing learning ability. Because of the parallel convolutional pathways, the traditional MSCNN, however, has a large number of parameters, which means a higher computing cost. In addition, a lack of consideration of contribution of different scale of degradation feature makes poor performance of RUL prediction. To address the issue, a pyramid CNN (PCNN) is proposed for RUL prediction of the milling tool is proposed in this paper. In this network, group convolution is used to replace parallel convolutional pathways to extract multiscale features without additional large number of parameters. The channel attention with soft assignment selects the key degradation features not only from different sensors but also from different scales. The proposed method was experimentally validated by the milling tool wear experiment. Some related methods and state-of-the-art models, including machine learning methods and deep learning methods, are analyzed for comparison with the proposed method. The result of it indicates that the proposed method is able to predict the RUL accurately.

Although the proposed method achieves a good RUL prediction result, there are still a few shortcomings in its application. The premise of the application of the proposed method is that the working condition of the testing data is the same as training data, which limits the application in practical engineering because the working condition of the machining process is dynamic. And limited labeled training samples prevents us from training a model for every working condition. To address the issue, a promising work is to introduce transfer learning or meta learning into the model, which can make the model achieve good performance under small samples. Furthermore, this can be combined with some adaptive optimization algorithms to automatically determine the hyperparameters of the model, which can achieve better performance of it.

Data Availability

The test data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This research was funded by the Key Research and Development Projects in Guangdong Province, grant number 2021B0101220005; Basic Research Program of Guangzhou, grant number 202102080349; industrial internet platform innovation development project of MIIT—New Potential Failure Mode and Effects Analysis System Based on Industrial Internet Platform, grant number TC210804F.

References

[1] Z. W. Lai, C. Y. Wang, L. J. Zheng et al., "Adaptability of AlTiN-based coated tools with green cutting technologies in sustainable machining of 316L stainless steel," *Tribology International*, vol. 148, Article ID 106300, 2020.

[2] W. Liu, W.-A. Yang, and Y. You, "Three-stage wiener-process-based model for remaining useful life prediction of a cutting tool in high-speed milling," *Sensors*, vol. 22, no. 13, p. 4763, 2022.

[3] Y. Lei, *Intelligent Fault Diagnosis and Remaining Useful Life Prediction of Rotating Machinery*, Elsevier Butterworth-Heinemann, Oxford, Amsterdam, Netherlands, 2016.

[4] C. C. Chen, B. Zhang, and G. Vachtsevanos, "Prediction of machine health condition using neuro-fuzzy and bayesian algorithms," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 2, pp. 297–306, 2012.

[5] P. Ding, M. P. Jia, and X. A. Yan, "Stationary subspaces-vector autoregressive with exogenous terms methodology for degradation trend estimation of rolling and slewing bearings," *Mechanical Systems and Signal Processing*, vol. 150, Article ID 107293, 2021.

[6] P. Park, M. Jung, and P. Di Marco, "Remaining useful life estimation of bearings using data-driven ridge regression," *Applied Sciences*, vol. 10, no. 24, p. 8977, 2020.

[7] M. H. Zhao, B. P. Tang, and Q. Tan, "Bearing remaining useful life estimation based on time-frequency representation and supervised dimensionality reduction," *Measurement*, vol. 86, pp. 41–55, 2016.

[8] X. Liu, Y. X. Jia, Z. W. He, and J. Zhou, "Application of EMD-WVD and particle filter for gearbox fault feature extraction and remaining useful life prediction," *Journal of Vibroengineering*, vol. 19, no. 3, pp. 1793–1808, 2017.

[9] B. Zhang, C. Sconyers, C. Byington, R. Patrick, M. E. Orchard, and G. Vachtsevanos, "A probabilistic fault detection approach: application to bearing fault detection," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 2011–2018, 2011.

[10] N. P. Li, Y. G. Lei, T. Yan, N. B. Li, and T. Y. Han, "A wiener-process-model-based method for remaining useful life prediction considering unit-to-unit variability," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 3, pp. 2092–2101, 2019.

[11] W. B. Chen, W. Z. Chen, H. X. Liu, Y. Q. Wang, C. L. Bi, and Y. Gu, "A RUL prediction method of small sample equipment based on DCNN-BiLSTM and domain adaptation," *Mathematics*, vol. 10, no. 7, p. 1022, 2022.

[12] H. Cheng, X. G. Kong, Q. B. Wang, H. B. Ma, S. K. Yang, and G. G. Chen, "Deep transfer learning based on dynamic domain adaptation for remaining useful life prediction under different working conditions," *Journal of Intelligent Manufacturing*, 2021.

[13] C. Sun, M. Ma, Z. B. Zhao, S. H. Tian, R. Q. Yan, and X. F. Chen, "Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2416–2425, 2019.

[14] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mechanical Systems and Signal Processing*, vol. 107, pp. 241–265, 2018.

[15] P. H. Li, X. Z. Liu, and Y. H. Yang, "Remaining useful life prognostics of bearings based on a novel spatial graph-temporal convolution network," *Sensors*, vol. 21, no. 12, p. 4217, 2021.

[16] L. X. Cao, Z. Qian, H. Zareipour et al., "Prediction of remaining useful life of wind turbine bearings under non-stationary operating conditions," *Energies*, vol. 11, no. 12, p. 3318, 2018.

[17] P. Ding and M. P. Jia, "Mechatronics equipment performance degradation assessment using limited and unlabeled data," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 4, pp. 2374–2385, 2022.

[18] H. Cheng, X. G. Kong, G. G. Chen, Q. B. Wang, and R. B. Wang, "Transferable convolutional neural network based remaining useful life prediction of bearing under multiple failure behaviors," *Measurement*, vol. 168, Article ID 108286, 2021.

[19] H. Cheng, X. G. Kong, Q. B. Wang, H. B. Ma, and S. K. Yang, "The Two-Stage RUL Prediction across Operation Conditions Using Deep Transfer Learning and Insufficient Degradation Data," *Reliability Engineering & System Safety*, vol. 225, Article ID 108581, 2022.

[20] P. Ding, M. P. Jia, J. C. Zhuang et al., "Multiobjective evolution enhanced collaborative health monitoring and prognostics: a case study of bearing life test with three-Axis acceleration signals," *IEEE Transactions on Instrumentation and Measurement*, vol. 71, pp. 71–12, 2022.

[21] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[22] F. Y. Deng, Y. Bi, Y. Q. Liu, and S. P. Yang, "Deep-learning-based remaining useful life prediction based on a multi-scale dilated convolution network," *Mathematics*, vol. 9, no. 23, p. 3035, 2021.

[23] B. Wang, Y. G. Lei, N. P. Li, and T. Yan, "Deep separable convolutional network for remaining useful life prediction of machinery," *Mechanical Systems and Signal Processing*, vol. 134, Article ID 106330, 2019.

[24] A. Z. Hinchi and M. Tkiouat, "Rolling element bearing remaining useful life estimation based on a convolutional long-short-term memory network," *Procedia Computer Science*, vol. 127, pp. 123–132, 2017.

[25] C. Zhang, P. Lim, A. K. Qin, and K. C. Tan, "Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2306–2318, 2017.

[26] C. Y. Wang, W. L. Jiang, X. K. Yang, and S. Q. Zhang, "RUL prediction of rolling bearings based on a DCAE and CNN," *Applied Sciences*, vol. 11, no. 23, Article ID 11516, 2021.

[27] P. Ding, M. P. Jia, Y. Ding, Y. Cao, X. Zhao, and X. Zhao, "Intelligent machinery health prognostics under variable operation conditions with limited and variable-length data," *Advanced Engineering Informatics*, vol. 53, Article ID 101691, 2022.

[28] P. Ding, M. P. Jia, and X. L. Zhao, "Meta deep learning based rotating machinery health prognostics toward few-shot prognostics," *Applied Soft Computing*, vol. 104, Article ID 107211, 2021.

[29] P. Ding, M. P. Jia, Y. F. Ding, and X. L. Zhao, "Statistical alignment-based metagated recurrent unit for cross-domain machinery degradation trend prognostics using limited data," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–12, 2021.

[30] W. Zhang, X. Li, H. Ma, Z. Luo, and X. Li, "Transfer learning using deep representation regularization in remaining useful life prediction across operating conditions," *Reliability Engineering & System Safety*, vol. 211, Article ID 107556, 2021.

[31] F. Y. Deng, H. Ding, S. P. Yang, and R. J. Hao, "An improved deep residual network with multiscale feature fusion for rotating machinery fault diagnosis," *Measurement Science and Technology*, vol. 32, no. 2, Article ID 024002, 2021.

[32] X. Li, W. Zhang, H. Ma, Z. Luo, and X. Li, "Degradation alignment in remaining useful life prediction using deep cycle-consistent learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 33, no. 10, pp. 5480–5491, 2022.

[33] H. Cheng, X. G. Kong, Q. B. Wang, H. B. Ma, and S. K. Yang, "The Two-Stage Rul Prediction across Operation Conditions Using Deep Transfer Learning and Insufficient Degradation Data," *Reliability Engineering & System Safety*, vol. 225, 2022.

[34] D. Z. Wu, C. Jennings, J. Terpenny, R. X. Gao, and S. Kumara, "A comparative study on machine learning algorithms for smart manufacturing: tool wear prediction using random forests," *Journal of Manufacturing Science and Engineering*, vol. 139, no. 7, 2017.

[35] X. Li, Q. Ding, and J. Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety*, vol. 172, pp. 1–11, 2018.

[36] Y. T. Li, Q. S. Xie, H. S. Huang, and Q. P. Chen, "Research on a tool wear monitoring algorithm based on residual dense network," *Symmetry*, vol. 11, no. 6, p. 809, 2019.

[37] H. Li, W. Zhao, Y. X. Zhang, and E. Zio, "Remaining useful life prediction using multi-scale deep convolutional neural network," *Applied Soft Computing*, vol. 89, Article ID 106113, 2020.

[38] Y. X. Chen, Y. Jin, and G. Jiri, "Predicting tool wear with multi-sensor data using deep belief networks," *International Journal of Advanced Manufacturing Technology*, vol. 99, no. 5-8, pp. 1917–1926, 2018.

[39] B. A. Wang, Y. G. Lei, N. P. Li, and W. T. Wang, "Multiscale convolutional attention network for predicting remaining useful life of machinery," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 8, pp. 7496–7504, 2021.